Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	1767	711/122,125,141,145.ccls.	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L2	44740	cache	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L3	199291	instruction?	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L4	7652	cache with instruction?	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L5	139350	command?	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L6	1783	cache with command?	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L7	8842	(cache with instruction?) or (cache with command?)	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L8	73734	tree	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L9	57643	trigger?	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L10	115409	age	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L11	80504	history	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L12	184393	age or history	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L13	274	(age or history) same trigger?	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L14	53	tree and ((age or history) same trigger?)	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L15	0	711/122,125,141,145.ccls. and (tree and ((age or history) same trigger?))	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L16	26	cache and (tree and ((age or history) same trigger?))	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L17	3211	deadlock	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51

L18	1042	cache and deadlock	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L19	107	711/122,125,141,145.ccls. and (cache and deadlock)	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L20	11	tree and (711/122,125,141,145.ccls. and (cache and deadlock))	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L21	1767	711/122,125,141,145.ccls.	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L22	44740	cache	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L23	199291	instruction?	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L24	7652	cache with instruction?	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L25	139350	command?	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L26	1783	cache with command?	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L27	8842	(cache with instruction?) or (cache with command?)	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L28	73734	tree	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L29	57643	trigger?	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L30	115409	age	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L31	80504	history	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L32	184393	age or history	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L33	274	(age or history) same trigger?	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L34	53	tree and ((age or history) same trigger?)	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L35	0	711/122,125,141,145.ccls. and (tree and ((age or history) same trigger?))	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51

L36	26	cache and (tree and ((age or history) same trigger?))	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L37	3211	deadlock	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L38	1042	cache and deadlock	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L39	107	711/122,125,141,145.ccls. and (cache and deadlock)	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L40	11	tree and (711/122,125,141,145.ccls. and (cache and deadlock))	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L41	3	"6119181".pn. or "6088795".pn. or "6081903". pn.	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L42	10	"6392912".pn. or "6389579".pn. or "6370596". pn. or "6349346".pn. or "6347346".pn. or "6341318".pn. or "6311200".pn. or "6288566". pn. or "6282627".pn. or "6243808".pn.	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L43	10	"6392912".pn. or "6389579".pn. or "6370596". pn. or "6349346".pn. or "6347346".pn. or "6341318".pn. or "6311200".pn. or "6288566". pn. or "6282627".pn. or "6243808".pn.	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L44	44740	cache	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L45	1	("6392912".pn. or "6389579".pn. or "6370596". pn. or "6349346".pn. or "6347346".pn. or "6341318".pn. or "6311200".pn. or "6288566". pn. or "6282627".pn. or "6243808".pn.) and cache	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L46	8842	(cache with instruction?) or (cache with command?)	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L47	46	(configurable adj cell? or configuration adj cell?) same (group or block)	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L48	376	(cache or caching or cached) with tree	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L49	1	((configurable adj cell? or configuration adj cell?) same (group or block)) and ((cache or caching or cached) with tree)	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L50	1	((cache with instruction?) or (cache with command?)) and ((configurable adj cell? or configuration adj cell?) same (group or block))	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L51	62	((cache with instruction?) or (cache with command?)) and ((cache or caching or cached) with tree)	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L52	1079938	configurable or configuration	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51

L53	23463	(configurable or configuration) with (cpu or processor or micro-processor or microprocessor or mp)	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L54	7	(((cache with instruction?) or (cache with command?)) and ((cache or caching or cached) with tree)) and ((configurable or configuration) with (cpu or processor or micro-processor or microprocessor or mp))	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L55	530	711/122.ccls.	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L56	229	711/125.ccls.	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L57	790	711/141.ccls.	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L58	528	711/145.ccls.	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L59	12711	(cache with instruction?) or (cache with command?)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L60	81	(configurable adj cell? or configuration adj cell?) same (group or block)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L61	765	(cache or caching or cached) with tree	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L62	1	((configurable adj cell? or configuration adj cell?) same (group or block)) and ((cache or caching or cached) with tree)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L63	1	((cache with instruction?) or (cache with command?)) and ((configurable adj cell? or configuration adj cell?) same (group or block))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L64	141	((cache with instruction?) or (cache with command?)) and ((cache or caching or cached) with tree)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L65	1400425	configurable or configuration	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L66	37181	(configurable or configuration) with (cpu or processor or micro-processor or microprocessor or mp)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L67	9	(((cache with instruction?) or (cache with command?)) and ((cache or caching or cached) with tree)) and ((configurable or configuration) with (cpu or processor or micro-processor or microprocessor or mp))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51

L68	14027	config$7 near4 cell?	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L69	2377	assign$4 near4 cach$3	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L70	1	(config$7 near4 cell?) same (assign$4 near4 cach$3)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L71	30	(assign$4 near4 cach$3) same tree	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L72	4	(config$7 near4 cell?) and ((assign$4 near4 cach$3) same tree)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L73	654	711/122.ccls.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L74	287	711/125.ccls.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L75	1000	711/141.ccls.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L76	676	711/145.ccls.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L77	7652	cache with instruction?	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L78	1783	cache with command?	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L79	53	tree and ((age or history) same trigger?)	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L80	0	(cache with instruction?) and (tree and ((age or history) same trigger?))	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L81	19	(cache with command?) and (tree and ((age or history) same trigger?))	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L82	141	((cache with instruction?) or (cache with command?)) and ((cache or caching or cached) with tree)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51

L83	4	(config$7 near4 cell?) and ((assign$4 near4 cach$3) same tree)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L84	9	(((cache with instruction?) or (cache with command?)) and ((cache or caching or cached) with tree)) and ((configurable or configuration) with (cpu or processor or micro-processor or microprocessor or mp))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L85	9268518	@ad<"19980225"	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L86	13	((config$7 near4 cell?) and ((assign$4 near4 cach$3) same tree)) or ((((cache with instruction?) or (cache with command?)) and ((cache or caching or cached) with tree)) and ((configurable or configuration) with (cpu or processor or micro-processor or microprocessor or mp)))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L87	5	@ad<"19980225" and (((config$7 near4 cell?) and ((assign$4 near4 cach$3) same tree)) or ((((cache with instruction?) or (cache with command?)) and ((cache or caching or cached) with tree)) and ((configurable or configuration) with (cpu or processor or micro-processor or microprocessor or mp))))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L88	48	configurable adj cell?	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L89	0	(@ad<"19980225" and (((config$7 near4 cell?) and ((assign$4 near4 cach$3) same tree)) or ((((cache with instruction?) or (cache with command?)) and ((cache or caching or cached) with tree)) and ((configurable or configuration) with (cpu or processor or micro-processor or microprocessor or mp))))) and (configurable adj cell?)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L90	0	@ad<"19980225" and ((((cache with instruction?) or (cache with command?)) and ((cache or caching or cached) with tree)) and (configurable adj cell?))	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L91	0	((cache with command?) and (tree and ((age or history) same trigger?))) and (configurable adj cell?)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L92	4	(((cache with instruction?) or (cache with command?)) and ((cache or caching or cached) with tree)) and (configurable adj cell?)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L93	15	reconfigurable adj cell?	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:59

L94	0	(((cache with instruction?) or (cache with command?)) and ((cache or caching or cached) with tree)) and (reconfigurable adj cell?)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L95	1767	711/122,125,141,145.ccls.	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L96	530	711/122.ccls.	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L97	229	711/125.ccls.	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L98	790	711/141.ccls.	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L99	528	711/145.ccls.	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L100	35	(configurable adj cell?) or (reconfigurable adj cell?)	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L101	2	((configurable adj cell?) or (reconfigurable adj cell?)) and 711/122,125,141,145.ccls.	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:53
L102	48	configurable adj cell?	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L103	8842	(cache with instruction?) or (cache with command?)	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L104	73734	tree	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51

L107	6	(((US-RE34363-$.DID. OR US-4591919-$.DID. OR US-4706216-$.DID. OR US-4739474-$.DID. OR US-4761755-$.DID. OR US-4811214-$.DID. OR US-4852048-$.DID. OR US-4860201-$.DID. OR US-4870302-$.DID. OR US-4901268-$.DID. OR US-4967340-$.DID. OR US-5014193-$.DID. OR US-5015884-$.DID. OR US-5021947-$.DID. OR US-5023775-$.DID. OR US-5043978-$.DID. OR US-5081375-$.DID. OR US-5109503-$.DID. OR US-5113498-$.DID. OR US-5115510-$.DID. OR US-5123109-$.DID. OR US-5125801-$.DID. OR US-5128559-$.DID. OR US-5142469-$.DID. OR US-5204935-$.DID. OR US-5208491-$.DID. OR US-5226122-$.DID. OR US-5233539-$.DID.) or (US-5247689-$.DID. OR US-5287472-$.DID. OR US-5301344-$.DID. OR US-5303172-$.DID. OR US-5336950-$.DID. OR US-5361373-$.DID. OR US-5418952-$.DID. OR US-5421019-$.DID. OR US-5422823-$.DID. OR US-5426378-$.DID. OR US-5430687-$.DID. OR US-5440245-$.DID. OR US-5440538-$.DID. OR US-5442790-$.DID. OR US-5444394-$.DID. OR US-5448186-$.DID. OR US-5455525-$.DID. OR US-5457644-$.DID. OR US-5473266-$.DID. OR US-5473267-$.DID. OR US-5475583-$.DID. OR US-5475803-$.DID. OR US-5483620-$.DID. OR US-0548510-$.DID. OR US-5485104-$.DID. OR US-0548985-$.DID. OR US-0549135-$.DID. OR US-0549323-$.DID. OR US-0549749-$.DID. OR US-0550699-$.DID. OR US-5510730-$.DID. OR US-5511173-$.DID. OR US-0551336-$.DID. OR US-0552183-$.DID. OR US-0552208-$.DID. OR US-0543269-$.DID. OR US-0553295-$.DID. OR US-0553540-$.DID.) or (US-5537057-$.DID. OR US-5537601-$.DID. OR US-5541530-$.DID. OR US-5544336-$.DID. OR US-5548773-$.DID. OR US-5555434-$.DID. OR US-5559450-$.DID. OR US-5561738-$.DID. OR US-5570040-$.DID. OR US-5583450-$.DID. OR US-5586044-$.DID. OR US-5587921-$.DID. OR US-5588152-$.DID. OR US-5590345-$.DID. OR US-5590348-$.DID. OR US-5596742-$.DID. OR US-5617547-$.DID. OR US-5634131-$.DID. OR US-5652894-$.DID. OR US-5655124-$.DID. OR US-5659797-$.DID. OR US-5713037-$.DID. OR US-5717943-$.DID. OR US-5734921-$.DID. OR US-5742180-$.DID. OR US-5748872-$.DID. OR US-5754871-$.DID. OR US-5761484-$.DID. OR US-5778439-$.DID. OR US-5801715-$.DID. OR US-5828858-$.DID. OR US-5838165-$.DID. OR US-5844888-$.DID. OR US-5867691-$.DID. OR US-5892961-$.DID. OR US-5915123-$.DID. OR US-5927423-$.DID. OR US-5936424-$.DID.) or (US-5956518-$.DID. OR US-6014509-$.DID. OR US-6052773-$.DID. OR US-6054873-$.DID. OR US-6108760-$.DID. OR US-6122719-$.DID. OR US-6127908-$.DID. OR US-5294119-$.DID. OR US-5611049-$.DID. OR US-5943242-$.DID. OR US-6081903-$.DID. OR US-6021490-$.DID. OR US-6038650-$.DID. OR US-6088795-$.DID. OR US-6119181-$.DID.)) and (configurable adj cell?)) or (((US-RE34363-$.DID. OR US-4591919-$.DID. OR US-4706216-$.DID. OR US-4739474-$.DID. OR US-4761755-$.DID. OR US-4811214-$.DID. OR US-4852048-$.DID. OR US-4860201-$.DID. OR US-4870302-$.DID. OR US-4901268-$.DID. OR US-4967340-$.DID. OR US-5014193-$.DID. OR US-5015884-$.DID. OR US-5021947-$.DID. OR US-5023775-$.DID. OR	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51

L108	28	US-RE34363-$.DID. OR US-4591979-$.DID. OR US-4706216-$.DID. OR US-4739474-$.DID. OR US-4761755-$.DID. OR US-4811214-$.DID. OR US-4852048-$.DID. OR US-4860201-$.DID. OR US-4870302-$.DID. OR US-4901268-$.DID. OR US-4967340-$.DID. OR US-5014193-$.DID. OR US-5015884-$.DID. OR US-5021947-$.DID. OR US-5023775-$.DID. OR US-5043978-$.DID. OR US-5081375-$.DID. OR US-5109503-$.DID. OR US-5113498-$.DID. OR US-5115510-$.DID. OR US-5123109-$.DID. OR US-5125801-$.DID. OR US-5128559-$.DID. OR US-5142469-$.DID. OR US-5204935-$.DID. OR US-5208491-$.DID. OR US-5226122-$.DID. OR US-5233539-$.DID.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L109	38	US-5247689-$.DID. OR US-5287472-$.DID. OR US-5301344-$.DID. OR US-5303172-$.DID. OR US-5336950-$.DID. OR US-5361373-$.DID. OR US-5418952-$.DID. OR US-5421019-$.DID. OR US-5422823-$.DID. OR US-5426378-$.DID. OR US-5430687-$.DID. OR US-5440245-$.DID. OR US-5440538-$.DID. OR US-5442790-$.DID. OR US-5444394-$.DID. OR US-5448186-$.DID. OR US-5455525-$.DID. OR US-5457644-$.DID. OR US-5473266-$.DID. OR US-5473267-$.DID. OR US-5475583-$.DID. OR US-5475803-$.DID. OR US-5483620-$.DID. OR US-5485103-$.DID. OR US-5485104-$.DID. OR US-5489857-$.DID. OR US-5491353-$.DID. OR US-5493239-$.DID. OR US-5497498-$.DID. OR US-5506998-$.DID. OR US-5510730-$.DID. OR US-5511173-$.DID. OR US-5513366-$.DID. OR US-5521837-$.DID. OR US-5522083-$.DID. OR US-5532693-$.DID. OR US-5532957-$.DID. OR US-5535406-$.DID.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L110	38	US-5537057-$.DID. OR US-5537601-$.DID. OR US-5541530-$.DID. OR US-5544336-$.DID. OR US-5548773-$.DID. OR US-5555434-$.DID. OR US-5559450-$.DID. OR US-5561738-$.DID. OR US-5570040-$.DID. OR US-5583450-$.DID. OR US-5586044-$.DID. OR US-5587921-$.DID. OR US-5588152-$.DID. OR US-5590345-$.DID. OR US-5590348-$.DID. OR US-5596742-$.DID. OR US-5617547-$.DID. OR US-5634131-$.DID. OR US-5652894-$.DID. OR US-5655124-$.DID. OR US-5659797-$.DID. OR US-5713037-$.DID. OR US-5717943-$.DID. OR US-5734921-$.DID. OR US-5742180-$.DID. OR US-5748872-$.DID. OR US-5754871-$.DID. OR US-5761484-$.DID. OR US-5778439-$.DID. OR US-5801715-$.DID. OR US-5828858-$.DID. OR US-5838165-$.DID. OR US-5844888-$.DID. OR US-5867691-$.DID. OR US-5892961-$.DID. OR US-5915123-$.DID. OR US-5927423-$.DID. OR US-5936424-$.DID.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51
L111	15	US-5956518-$.DID. OR US-6014509-$.DID. OR US-6052773-$.DID. OR US-6054873-$.DID. OR US-6108760-$.DID. OR US-6122719-$.DID. OR US-6127908-$.DID. OR US-5294119-$.DID. OR US-5611049-$.DID. OR US-5943242-$.DID. OR US-6081903-$.DID. OR US-6021490-$.DID. OR US-6038650-$.DID. OR US-6088795-$.DID. OR US-6119181-$.DID.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51

L112	119	(US-RE34363-$.DID. OR US-4591979-$.DID. OR US-4706216-$.DID. OR US-4739474-$.DID. OR US-4761755-$.DID. OR US-4811214-$.DID. OR US-4852048-$.DID. OR US-4860201-$.DID. OR US-4870302-$.DID. OR US-4901268-$.DID. OR US-4967340-$.DID. OR US-5014193-$.DID. OR US-5015884-$.DID. OR US-5021947-$.DID. OR US-5023775-$.DID. OR US-5043978-$.DID. OR US-5081375-$.DID. OR US-5109503-$.DID. OR US-5113498-$.DID. OR US-5115510-$.DID. OR US-5123109-$.DID. OR US-5125801-$.DID. OR US-5128559-$.DID. OR US-5142469-$.DID. OR US-5204935-$.DID. OR US-5208491-$.DID. OR US-5226122-$.DID. OR US-5233539-$.DID.) or (US-5247689-$.DID. OR US-5287472-$.DID. OR US-5301344-$.DID. OR US-5303172-$.DID. OR US-5336950-$.DID. OR US-5361373-$.DID. OR US-5418952-$.DID. OR US-5421019-$.DID. OR US-5422823-$.DID. OR US-5426378-$.DID. OR US-5430687-$.DID. OR US-5440245-$.DID. OR US-5440538-$.DID. OR US-5442790-$.DID. OR US-5444394-$.DID. OR US-5448186-$.DID. OR US-5455525-$.DID. OR US-5457644-$.DID. OR US-5473266-$.DID. OR US-5473267-$.DID. OR US-5475583-$.DID. OR US-5475803-$.DID. OR US-5483620-$.DID. OR US-5485103-$.DID. OR US-5485104-$.DID. OR US-5489857-$.DID. OR US-5491353-$.DID. OR US-5493239-$.DID. OR US-5497498-$.DID. OR US-5506998-$.DID. OR US-5510730-$.DID. OR US-5511173-$.DID. OR US-5513366-$.DID. OR US-5521837-$.DID. OR US-5522083-$.DID. OR US-5532693-$.DID. OR US-5532957-$.DID. OR US-5535406-$.DID.) or (US-5537057-$.DID. OR US-5537601-$.DID. OR US-5541530-$.DID. OR US-5544336-$.DID. OR US-5548773-$.DID. OR US-5555434-$.DID. OR US-5559450-$.DID. OR US-5561738-$.DID. OR US-5570040-$.DID. OR US-5583450-$.DID. OR US-5586044-$.DID. OR US-5587921-$.DID. OR US-5588152-$.DID. OR US-5590345-$.DID. OR US-5590348-$.DID. OR US-5596742-$.DID. OR US-5617547-$.DID. OR US-5634131-$.DID. OR US-5652894-$.DID. OR US-5655124-$.DID. OR US-5659797-$.DID. OR US-5713037-$.DID. OR US-5717943-$.DID. OR US-5734921-$.DID. OR US-5742180-$.DID. OR US-5748872-$.DID. OR US-5754871-$.DID. OR US-5761484-$.DID. OR US-5778439-$.DID. OR US-5801715-$.DID. OR US-5828858-$.DID. OR US-5838165-$.DID. OR US-5844888-$.DID. OR US-5867691-$.DID. OR US-5892961-$.DID. OR US-5915123-$.DID. OR US-5927423-$.DID. OR US-5936424-$.DID.) or (US-5956518-$.DID. OR US-6014509-$.DID. OR US-6052773-$.DID. OR US-6054873-$.DID. OR US-6108760-$.DID. OR US-6122719-$.DID. OR US-6127908-$.DID. OR US-5294119-$.DID. OR US-5611049-$.DID. OR US-5943242-$.DID. OR US-6081903-$.DID. OR US-6021490-$.DID. OR US-6038650-$.DID. OR US-6088795-$.DID. OR US-6119181-$.DID.)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51

L113	115441	(configurable adj cell?) or ((cache with instruction?) or (cache with command?)) or tree	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51

L114	32	((US-RE34363-$.DID. OR US-4591979-$.DID. OR US-4706216-$.DID. OR US-4739474-$.DID. OR US-4761755-$.DID. OR US-4811214-$.DID. OR US-4852048-$.DID. OR US-4860201-$.DID. OR US-4870302-$.DID. OR US-4901268-$.DID. OR US-4967340-$.DID. OR US-5014193-$.DID. OR US-5015884-$.DID. OR US-5021947-$.DID. OR US-5023775-$.DID. OR US-5043978-$.DID. OR US-5081375-$.DID. OR US-5109503-$.DID. OR US-5113498-$.DID. OR US-5115510-$.DID. OR US-5123109-$.DID. OR US-5125801-$.DID. OR US-5128559-$.DID. OR US-5142469-$.DID. OR US-5204935-$.DID. OR US-5208491-$.DID. OR US-5226122-$.DID. OR US-5233539-$.DID.) or (US-5247689-$.DID. OR US-5287472-$.DID. OR US-5301344-$.DID. OR US-5303172-$.DID. OR US-5336950-$.DID. OR US-5361373-$.DID. OR US-5418952-$.DID. OR US-5421019-$.DID. OR US-5422823-$.DID. OR US-5426378-$.DID. OR US-5430687-$.DID. OR US-5440245-$.DID. OR US-5440538-$.DID. OR US-5442790-$.DID. OR US-5444394-$.DID. OR US-5448186-$.DID. OR US-5455525-$.DID. OR US-5457644-$.DID. OR US-5473266-$.DID. OR US-5473267-$.DID. OR US-5475583-$.DID. OR US-5475803-$.DID. OR US-5483620-$.DID. OR US-5485103-$.DID. OR US-5485104-$.DID. OR US-5489857-$.DID. OR US-5491353-$.DID. OR US-5493239-$.DID. OR US-5497498-$.DID. OR US-5506998-$.DID. OR US-5510730-$.DID. OR US-5511173-$.DID. OR US-5513366-$.DID. OR US-5521837-$.DID. OR US-5522083-$.DID. OR US-5532693-$.DID. OR US-5532957-$.DID. OR US-5535406-$.DID.) or (US-5537057-$.DID. OR US-5537601-$.DID. OR US-5541530-$.DID. OR US-5544336-$.DID. OR US-5548773-$.DID. OR US-5555434-$.DID. OR US-5559450-$.DID. OR US-5561738-$.DID. OR US-5570040-$.DID. OR US-5583450-$.DID. OR US-5586044-$.DID. OR US-5587921-$.DID. OR US-5588152-$.DID. OR US-5590345-$.DID. OR US-5590348-$.DID. OR US-5596742-$.DID. OR US-5617547-$.DID. OR US-5634131-$.DID. OR US-5652894-$.DID. OR US-5655124-$.DID. OR US-5659797-$.DID. OR US-5713037-$.DID. OR US-5717943-$.DID. OR US-5734921-$.DID. OR US-5742180-$.DID. OR US-5748872-$.DID. OR US-5754871-$.DID. OR US-5761484-$.DID. OR US-5778439-$.DID. OR US-5801715-$.DID. OR US-5828858-$.DID. OR US-5838165-$.DID. OR US-5844888-$.DID. OR US-5867691-$.DID. OR US-5892961-$.DID. OR US-5915123-$.DID. OR US-5927423-$.DID. OR US-5936424-$.DID.) or (US-5956518-$.DID. OR US-6014509-$. DID. OR US-6052773-$.DID. OR US-6054873-$. DID. OR US-6108760-$.DID. OR US-6122719-$. DID. OR US-6127908-$.DID. OR US-5294119-$. DID. OR US-5611049-$.DID. OR US-5943242-$. DID. OR US-6081903-$.DID. OR US-6021490-$. DID. OR US-6038650-$.DID. OR US-6088795-$. DID. OR US-6119181-$.DID.)) and ((configurable adj cell?) or ((cache with instruction?) or (cache with command?)) or tree)	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:51

L115	119	114 or 112	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:54
L116	5	102 and 115	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:58
L117	2	93 and 115	USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:59
L118	2	93 same instruction	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/05/11 15:59

e-mail printer friendly

Results for "(((('reconfigurable cell' or fgpa) and (instruction or command))<in>metadata)"

Your search matched 0 of 1157693 documents.

A maximum of 100 results are displayed, 25 to a page, sorted by Relevance in Descending order.

» View Session History

» New Search

Modify Search

(((('reconfigurable cell' or fgpa) and (instruction or command))<in>metadata)

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistance revising your search.

e-mail printer friendly

Results for "(('reconfigurable cell' and 'reconfiguration instruction')<in>metadata)"

Your search matched **0** of **1157693** documents.

A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» View Session History

» New Search

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

Modify Search

((reconfigurable cell' and 'reconfiguration instruction')<in>metadata)

Check to search only within this results set

Display Format: Citation Citation & Abstract

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistance revising your search.

Results for "(reconfigurable and cell and (instruction or command)<in>metadata)"

e-mail printer friendly

Your search matched 161 of 1157693 documents.

A maximum of 100 results are displayed, 25 to a page, sorted by Relevance in Descending order.

» View Session History

» New Search

Modify Search

(reconfigurable and cell and (instruction or command)<in>metadata) >>

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

Select Article Information

View: 1-25 | 26-50 | 51-75 | 76-100

1. **A VLIW processor with reconfigurable instruction set for embedded applications**
Lodi, A.; Toma, M.; Campi, F.; Cappelli, A.; Canegallo, R.; Guerrieri, R.;
Solid-State Circuits, IEEE Journal of
Volume 38, Issue 11, Nov. 2003 Page(s):1876 - 1886
AbstractPlus | References | Full Text: PDF(1712 KB) IEEE JNL

2. **Development support software of cell array processor and its neural network application**
Akita, N.; Hatano, F.; Komoku, K.; Morishita, T.; Teramoto, I.;
Neural Networks, 1999. IJCNN '99. International Joint Conference on
Volume 4, 10-16 July 1999 Page(s):2452 - 2455 vol.4
AbstractPlus | Full Text: PDF(356 KB) IEEE CNF

3. **MPEG-compliant entropy decoding on FPGA-augmented TriMedia/CPU64**
Simat, M.; Cotofana, S.; Vassiliadis, S.; van Eijndhoven, J.T.J.; Vissers, K.;
Field-Programmable Custom Computing Machines, 2002. Proceedings. 10th Annual IEEE Symposium on
22-24 April 2002 Page(s):261 - 270
AbstractPlus | Full Text: PDF(1934 KB) IEEE CNF

4. **A heterogeneous multiprocessor architecture for low-power audio signal processing applications**
Paker, O.; Sparso, J.; Haandbaek, N.; Isager, M.; Nielsen, L.S.;
VLSI, 2001. Proceedings. IEEE Computer Society Workshop on
19-20 April 2001 Page(s):47 - 53
AbstractPlus | Full Text: PDF(876 KB) IEEE CNF

5. **Low-power control architecture for embedded processors**
Mattos, J.C.B.; Kreutz, M.; Carro, L.;
Integrated Circuits and Systems Design, 2002. Proceedings. 15th Symposium on
9-14 Sept. 2002 Page(s):221 - 226
AbstractPlus | Full Text: PDF(265 KB) IEEE CNF

6. **Improving the operation autonomy of SIMD processing elements by using guarded instructions and pseudo branches**
Anido, M.L.; Paar, A.; Bagherzadeh, N.;
Digital System Design, 2002. Proceedings. Euromicro Symposium on
4-6 Sept. 2002 Page(s):148 - 155
AbstractPlus | Full Text: PDF(469 KB) IEEE CNF

7. **Stream PRAM**
Ulm, D.R.; Scherger, M.;
Parallel and Distributed Processing Symposium, 2005. Proceedings. 19th IEEE International
04-08 April 2005 Page(s):265a - 265a
AbstractPlus | Full Text: PDF(184 KB) IEEE CNF

8. **A reconfigurable approach to a systolic sorting architecture**
Alexander, T.; Soma, M.;
Circuits and Systems, 1989., IEEE International Symposium on
8-11 May 1989 Page(s):1178 - 1182 vol.2

AbstractPlus | Full Text: PDF(384 KB) IEEE CNF

9. **A configurable architecture for smart pixel research**
Cantin, J.F.; Beyette, F.R., Jr.;
Electronic-Enhanced Optics, Optical Sensing in Semiconductor Manufacturing, Electro-Optics in Space, Broadband Optical Networks, 2000. Digest of the LEOS Summer Topical Meetings
24-28 July 2000 Page(s):I67 - I68

AbstractPlus | Full Text: PDF(256 KB) IEEE CNF

10. **A single-chip, 1.6-billion, 16-b MAC/s multiprocessor DSP**
Ackland, B.; Anesko, A.; Brinthaupt, D.; Daubert, S.J.; Kalavade, A.; Knobloch, J.; Micca, E.; Moturi, M.; Nicol, C.J.; O'Neill, J.H.; Othmer, J.; Sackinger, E.; Singh, K.J.; Sweet, J.; Terman, C.J.; Williams, J.;
Solid-State Circuits, IEEE Journal of
Volume 35, Issue 3, March 2000 Page(s):412 - 424

AbstractPlus | References | Full Text: PDF(828 KB) IEEE JNL

11. **A novel method for improving the operation autonomy of SIMD processing elements**
Anido, M.L.; Paar, A.; Bagherzadeh, N.;
Integrated Circuits and Systems Design, 2002. Proceedings. 15th Symposium on
9-14 Sept. 2002 Page(s):49 - 54

AbstractPlus | Full Text: PDF(406 KB) IEEE CNF

12. **FPGA prototyping of a RISC processor core for embedded applications**
Gschwind, M.; Salapura, V.; Maurer, D.;
Very Large Scale Integration (VLSI) Systems, IEEE Transactions on
Volume 9, Issue 2, April 2001 Page(s):241 - 250

AbstractPlus | References | Full Text: PDF(436 KB) IEEE JNL

13. **The UCSC Kestrel parallel processor**
Di Bias, A.; Dahle, D.M.; Diekhans, M.; Grate, L.; Hirschberg, J.; Karplus, K.; Keller, H.; Kendrick, M.; Mesa-Martinez, F.J.; Pease, D.; Rice, E.; Schultz, A.; Speck, D.; Hughey, R.;
Parallel and Distributed Systems, IEEE Transactions on
Volume 16, Issue 1, Jan. 2005 Page(s):80 - 92

AbstractPlus | Full Text: PDF(1168 KB) IEEE JNL

14. **Tag compression for low power in dynamically customizable embedded processors**
Petrov, P.; Orailoglu, A.;
Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on
Volume 23, Issue 7, July 2004 Page(s):1031 - 1047

AbstractPlus | References | Full Text: PDF(560 KB) IEEE JNL

15. **ACE16k: the third generation of mixed-signal SIMD-CNN ACE chips toward VSoCs**
Rodriguez-Vazquez, A.; Linan-Cembrano, G.; Carranza, L.; Roca-Moreno, E.; Carmona-Galan, R.; Jimenez-Garrido, F.; Dominguez-Castro, R.; Meana, S.E.;
Circuits and Systems I: Regular Papers, IEEE Transactions on [see also Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on]
Volume 51, Issue 5, May 2004 Page(s):851 - 863

AbstractPlus | Full Text: PDF(656 KB) IEEE JNL

16. **The Chimaera reconfigurable functional unit**
Hauck, S.; Fry, T.W.; Hosler, M.M.; Kao, J.P.;
Very Large Scale Integration (VLSI) Systems, IEEE Transactions on
Volume 12, Issue 2, Feb. 2004 Page(s):206 - 217

AbstractPlus | References | Full Text: PDF(376 KB) IEEE JNL

17. **IP core implementation of a self-organizing neural network**
Hendry, D.C.; Duncan, A.A.; Lightowler, N.;
Neural Networks, IEEE Transactions on

Volume 14, Issue 5, Sept. 2003 Page(s):1085 - 1096

AbstractPlus | References | Full Text: PDF(524 KB) IEEE JNL

18. **A polymorphous computing fabric**
Wolinski, C.; Gokhale, M.; McCave, K.;
Micro, IEEE
Volume 22, Issue 5, Sept.-Oct. 2002 Page(s):56 - 68

AbstractPlus | References | Full Text: PDF(340 KB) IEEE JNL

19. **An energy-efficient reconfigurable public-key cryptography processor**
Goodman, J.; Chandrakasan, A.P.;
Solid-State Circuits, IEEE Journal of
Volume 36, Issue 11, Nov. 2001 Page(s):1808 - 1820

AbstractPlus | References | Full Text: PDF(440 KB) IEEE JNL

20. **Reducing leakage in a high-performance deep-submicron instruction cache**
Powell, M.; Se-Hyun Yang; Falsafi, B.; Roy, K.; Vijaykumar, N.;
Very Large Scale Integration (VLSI) Systems, IEEE Transactions on
Volume 9, Issue 1, Feb. 2001 Page(s):77 - 89

AbstractPlus | References | Full Text: PDF(228 KB) IEEE JNL

21. **A digital signal processor with programmable correlator array architecture for third generation wireless communication system**
Chi-Kuang Chen; Po-Chih Tseng; Yung-Chil Chang; Liang-Gee Chen;
Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on [see also Circuits and Systems II: Express Briefs, IEEE Transactions on]
Volume 48, Issue 12, Dec. 2001 Page(s):1110 - 1120

AbstractPlus | References | Full Text: PDF(449 KB) IEEE JNL

22. **A separated bit-line unified cache: Conciliating small on-chip cache die-area and low miss ratio**
Mizuno, H.; Ishibashi, K.;
Very Large Scale Integration (VLSI) Systems, IEEE Transactions on
Volume 7, Issue 1, March 1999 Page(s):139 - 144

AbstractPlus | References | Full Text: PDF(124 KB) IEEE JNL

23. **ANT-on-YARDS: FPGA/MPU hybrid architecture for telecommunication data processing**
Tsutsui, A.; Miyazaki, T.;
Very Large Scale Integration (VLSI) Systems, IEEE Transactions on
Volume 6, Issue 2, June 1998 Page(s):199 - 211

AbstractPlus | References | Full Text: PDF(332 KB) IEEE JNL

24. **An embedded reconfigurable SIMD DSP with capability of dimension-controllable vector processing**
Liang Han; Jie Chen; Chaoxian Zhou; Ying Li; Xin Zhang; Zhibi Liu; Xiaoyun Wei; Baofeng Li;
Computer Design: VLSI in Computers and Processors, 2004. ICCD 2004. Proceedings. IEEE International Conference on
11-13 Oct. 2004 Page(s):446 - 451

AbstractPlus | Full Text: PDF(402 KB) IEEE CNF

25. **Application of VLSI for image processing**
Jaffe, R.C.; Johnson, D.S.; Lin, W.-T.; Ho, C.-Y.;
Acoustics, Speech, and Signal Processing, 1989. ICASSP-89., 1989 International Conference on
23-26 May 1989 Page(s):2421 - 2424 vol.4

AbstractPlus | Full Text: PDF(280 KB) IEEE CNF

View: 1-25 | 26-50 | 51-75 | 76-100

Results for "('reconfigurable cell' and (instruction or command)<in>metadata)"

e-mail printer friendly

Your search matched 6 of 1157693 documents.

A maximum of 100 results are displayed, 25 to a page, sorted by Relevance in Descending order.

» View Session History

» New Search

Modify Search

('reconfigurable cell' and (instruction or command)<in>metadata)

» Key

IEEE JNL IEEE Journal or Magazine

IEE JNL IEE Journal or Magazine

IEEE CNF IEEE Conference Proceeding

IEE CNF IEE Conference Proceeding

IEEE STD IEEE Standard

Check to search only within this results set

Display Format: Citation Citation & Abstract

Select Article Information

1. **Improving the operation autonomy of SIMD processing elements by using guarded instructions and pseudo branches**
Anido, M.L.; Paar, A.; Bagherzadeh, N.;
Digital System Design, 2002. Proceedings. Euromicro Symposium on
4-6 Sept. 2002 Page(s):148 - 155
AbstractPlus | Full Text: PDF(469 KB) IEEE CNF

2. **A VLIW processor with reconfigurable instruction set for embedded applications**
Lodi, A.; Toma, M.; Campi, F.; Cappelli, A.; Canegallo, R.; Guerrieri, R.;
Solid-State Circuits, IEEE Journal of
Volume 38, Issue 11, Nov. 2003 Page(s):1876 - 1886
AbstractPlus | References | Full Text: PDF(1712 KB) IEEE JNL

3. **The MorphoSys dynamically reconfigurable system-on-chip**
Guangming Lu; Singh, H.; Ming-Hau Lee; Bagherzadeh, N.; Kurdahi, F.J.; Filho, E.M.C.; Castro-Alves, V.;
Evolvable Hardware, 1999. Proceedings of the First NASA/DoD Workshop on
19-21 July 1999 Page(s):152 - 160
AbstractPlus | Full Text: PDF(708 KB) IEEE CNF

4. **Development of a test environment for pre- and post-synthesis verification of correct VHDL description of core processor systems**
Heath, J.R.; Stroud, C.E.; Leong, W.D.;
Fall VIUF Workshop, 1999.
4-6 Oct. 1999 Page(s):40 - 46
AbstractPlus | Full Text: PDF(72 KB) IEEE CNF

5. **A novel method for improving the operation autonomy of SIMD processing elements**
Anido, M.L.; Paar, A.; Bagherzadeh, N.;
Integrated Circuits and Systems Design, 2002. Proceedings. 15th Symposium on
9-14 Sept. 2002 Page(s):49 - 54
AbstractPlus | Full Text: PDF(406 KB) IEEE CNF

6. **Design methodology for a tightly coupled VLIW/reconfigurable matrix architecture: a case study**
Mei, B.; Vernalde, S.; Verkest, D.; Lauwereins, R.;
Design, Automation and Test in Europe Conference and Exhibition, 2004. Proceedings
Volume 2, 16-20 Feb. 2004 Page(s):1224 - 1229 Vol.2
AbstractPlus | Full Text: PDF(287 KB) IEEE CNF

Terms used **reconfigurable cell** and **instruction** or **command**

Found **9,346** of **154,226**

Sort results by relevance
Display results expanded form
Save results to a Binder
Search Tips
Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next
Best 200 shown

Relevance scale

1 Beyond command knowledge: identifying and teaching strategic knowledge for using complex computer applications
Suresh K. Bhavnani, Frederick Reif, Bonnie E. John
March 2001 **Proceedings of the SIGCHI conference on Human factors in computing systems**
Full text available: pdf(245.62 KB) Additional Information: full citation, abstract, references, index terms

Despite experience, many users do not make efficient use of complex computer applications. We argue that this is caused by a lack of strategic knowledge that is difficult to acquire just by knowing how to use commands. To address this problem, we present efficient and general strategies for using computer applications, and identify the components of strategic knowledge required to use them. We propose a framework for teaching strategic knowledge, and show how we implemented it in a course f ...

Keywords: GOMS, instruction, strategies, training

2 From algorithm parallelism to instruction-level parallelism: an encode-decode chain using prefix-sum
Uzi Vishkin
June 1997 **Proceedings of the ninth annual ACM symposium on Parallel algorithms and architectures**
Full text available: pdf(1.68 MB) Additional Information: full citation, references, citings

3 THE SITE MACHINE Computer-aided instruction in architectural education
Edward F. Smith
January 1977 **Proceedings of the 14th conference on Design automation**
Full text available: pdf(763.61 KB) Additional Information: full citation, abstract, references, index terms

THE SITE MACHINE is one of two prototype, graphics-based programs written to evaluate the potential of computer-aided instruction (CAI) programs in architectural education. Some CAI systems are now highly developed but almost no use has occurred in architectural education. The special characteristics and diverse nature of architectural education as well as the requirement for fast, high quality graphics, especially in the design areas, requires special applications and consideration. ...

4 A microprogramming simulator for instructional use
J. R. Parker, K. Becker
January 1984 **ACM SIGCSE Bulletin , Proceedings of the fifteenth SIGCSE technical symposium on Computer science education**, Volume 16 Issue 1
Full text available: pdf(598.12 KB) Additional Information: full citation, abstract, references, citings, index terms

The teaching of computer architecture at a low level is made difficult by the complexity of the real systems which are used as examples and tools. This paper describes a processor simulation system which is intended for use at the second and third year undergraduate level for teaching techniques and concepts in the implementation of instruction sets and microprogramming. The important features of this system are in the user interface, and not necessarily in the actual processor which is sim ...

5 Explicit multi-threading (XMT) bridging models for instruction parallelism (extended abstract)
Uzi Vishkin, Shlomit Dascal, Efraim Berkovich, Joseph Nuzman
June 1998 **Proceedings of the tenth annual ACM symposium on Parallel algorithms and architectures**
Full text available: pdf(1.71 MB) Additional Information: full citation, references, citings, index terms

6 Speech synthesis for computer assisted instruction: The MISS system and its applications
William R. Sanders, Gerard V. Benbassat, Robert L. Smith
February 1976 **Proceedings of the ACM SIGCSE-SIGCUE technical symposium on Computer science and education**, Volume 2 , 8 Issue SI , 1
Full text available: pdf(1.03 MB) Additional Information: full citation, abstract, references, index terms

The Institute for Mathematical Studies in the Social Sciences at Stanford (IMSSS) has developed a synthesis system, MISS (Microprogrammed Intoned Speech Synthesizer), designed to test the effectiveness of computer-generated speech in the context of complex CAI programs. No one method of computer controlled speech production is completely satisfactory for all the uses of computer-assisted instruction (CAI). The choice of synthesis method is strongly related to the kinds of curriculums and in ...

7 Some experiments in man-machine interaction relevant to computer assisted instruction
Philip G. Barker
January 1981 **Proceedings of the ACM '81 conference**
Full text available: pdf(748.21 KB) Additional Information: full citation, abstract, references, index terms

Some experiments in man-machine interaction are briefly described in the context of their relevance to multi-media audio-visual instruction. Two important points emerge: the need for the development of appropriate data base management systems and the necessity for a suitable author language capable of providing the instructor with a means of coordinating the use of a wide variety of instructional resources.

8 The impact of menus and command-level feedback on learners' acquisition of data base language skills
Mary Sumner, James Benjamin
February 1988 **ACM SIGCSE Bulletin , Proceedings of the nineteenth SIGCSE technical symposium on Computer science education**, Volume 20 Issue 1
Full text available: pdf(580.54 KB) Additional Information: full citation, abstract, references, index terms

The purpose of this study was to determine whether providing menus and postactive feedback of command-level syntax can facilitate the acquisition of formal language skills by novice learners. Two groups of students, one of which received training in a menu version of dBASE III Plus and the other of which received instruction in a command version, were asked to complete data base file maintenance and query tasks. Measures comparing the performance of the two groups on a post-test were comple ...

9 CG-1, a course generating program for computer-assisted instruction
Charles T Meadow, Douglas W Waugh, Forrest E Miller
January 1968 **Proceedings of the 1968 23rd ACM national conference**
Full text available: pdf(838.80 KB) Additional Information: full citation, abstract, references, index terms

In this paper we discuss a computer program which assists authors in producing computer-assisted instruction courses. The courses are executable computer programs. The course generator, CG-1, is an interactive program which produces a course program as the result of a conversation between the computer and the course author carried on entirely in natural language. The generated programs are in a language called the PL/I Interactive Dialect (PL/I ID) which is derived from PL/I. CG-1 was writt ...

10 Natural command names and initial learning: a study of text-editing terms
T. K. Landauer, K. M. Galotti, S. Hartwell
July 1983 **Communications of the ACM**, Volume 26 Issue 7
Full text available: pdf(1.06 MB) Additional Information: full citation, abstract, references, citings, index terms

In the first of two studies of "naturalness" in command names, computer-naive typists composed instructions to "someone else" for correcting a sample text. There was great variety in their task-descriptive lexicon and a lack of correspondence between both their vocabulary and their underlying conceptions of the editing operations and those of some computerized text editors. In the second study, computer-naive typists spent two hours learning minimal text-edit ...

Keywords: dialogue, human-computer interaction, user interfaces

11 Experiments with list ranking for explicit multi-threaded (XMT) instruction parallelism
Dascal Vishkin, Uzi Vishkin
December 2000 **Journal of Experimental Algorithmics (JEA)**, Volume 5
Full text available: pdf(347.52 KB) ps(382.86 KB) Additional Information: full citation, abstract, references, citings, index terms

Algorithms for the problem of list ranking are empirically studied with respect to the Explicit Multi-Threaded (XMT) platform for instruction-level parallelism (ILP). The main goal of this study is to understand the differences between XMT and more traditional parallel computing implementation platforms/models as they pertain to the well studied list ranking problem. The main two findings are: (i) good speedups for much smaller inputs are possible and (ii) in part, the first finding i ...

12 Military applications: An agent architecture for implementing command and control in military simulations

Colin R. Mason, James Moffat

December 2001 **Proceedings of the 33nd conference on Winter simulation**

Full text available: pdf(255.07 KB) Additional Information: full citation, abstract, references, citings, index terms

In models of military operations it is important to include the Command and Control (C2) process in order to achieve a realistic simulation of a military force's behaviour and effectiveness. Inspired by ideas from complexity theory we have developed a representation of C2 based on a decentralised system of interacting intelligent "command agents". In this paper we describe the architecture of our command agents and how this captures the key C2 processes that exist in military headquarters, parti ...

13 Commands as media: design and implementation of a command stream

Jonathan L. Herlocker, Joseph A. Konstan

January 1995 **Proceedings of the third ACM international conference on Multimedia**

Full text available: htm(47.29 KB) Additional Information: full citation, references, citings, index terms

Keywords: TclStream, command stream, commands, multimedia presentations, reversibilty

14 What kind of minimal instruction manual is the most effective

John B. Black, John M. Carroll, Stuart M. McGuigan

May 1986 **ACM SIGCHI Bulletin , Proceedings of the SIGCHI/GI conference on Human factors in computing systems and graphics interface**, Volume 17 Issue SI

Full text available: pdf(371.03 KB) Additional Information: full citation, abstract, references, citings, index terms

An empirical study examined the effectiveness of four different versions of a self-instruction manual for a word processing system: a Skeletal version that explicitly states only the essential information, an Inferential version that has the users infer some of the essential information, a Rehearsal version that is like the Skeletal manual, but adds opportunities to rehearse the explicitly stated information, and a Lengthy version that adds nonessential explanatory and descriptive informati ...

Keywords: instruction, learning, manual design

15 New techniques for presenting instructions and transcripts: Comparative effectiveness of augmented reality in object assembly

Arthur Tang, Charles Owen, Frank Biocca, Weimin Mou

April 2003 **Proceedings of the SIGCHI conference on Human factors in computing systems**

Full text available: pdf(237.22 KB) Additional Information: full citation, abstract, references, citings, index terms

Although there has been much speculation about the potential of Augmented Reality (AR), there are very few empirical studies about its effectiveness. This paper describes an experiment that tested the relative effectiveness of AR instructions in an assembly task. Task information was displayed in user's field of view and registered with the workspace as 3D objects to explicitly demonstrate the exact execution of a procedure step. Three instructional media were compared with the AR system: a prin ...

Keywords: augmented reality, computer assisted instruction, human computer interaction, usability study

16 Command abbreviation behavior in human-computer interaction

Izak Benbasat, Yair Wand

April 1984 **Communications of the ACM**, Volume 27 Issue 4

Full text available: pdf(611.32 KB) Additional Information: full citation, references, citings, index terms

Keywords: abbreviations, human-computer dialogue, interactive systems, user interfaces

17 Intrusion detection: Countering code-injection attacks with instruction-set randomization

Gaurav S. Kc, Angelos D. Keromytis, Vassilis Prevelakis

October 2003 **Proceedings of the 10th ACM conference on Computer and communications security**

Full text available: pdf(146.35 KB) Additional Information: full citation, abstract, references, citings, index terms

We describe a new, general approach for safeguarding systems against *any* type of code-injection attack. We

apply Kerckhoff's principle, by creating process-specific randomized instruction sets (*e.g.,* machine instructions) of the system executing potentially vulnerable software. An attacker who does not know the key to the randomization algorithm will inject code that is invalid for that randomized processor, causing a runtime exception. To determine the difficulty of integrating su ...

Keywords: buffer overflows, emulators, interpreters

18 The KScalar simulator

J. C. Moure, Dolores I. Rexachs, Emilio Luque

March 2002 **Journal on Educational Resources in Computing (JERIC)**, Volume 2 Issue 1

Full text available: pdf(493.35 KB) Additional Information: full citation, abstract, references, index terms

Modern processors increase their performance with complex microarchitectural mechanisms, which makes them more and more difficult to understand and evaluate. KScalar is a graphical simulation tool that facilitates the study of such processors. It allows students to analyze the performance behavior of a wide range of processor microarchitectures: from a very simple in-order, scalar pipeline, to a detailed out-of-order, superscalar pipeline with non-blocking caches, speculative execution, and comp ...

Keywords: Education, pipelined processor simulator

19 The structure and performance of interpreters

Theodore H. Romer, Dennis Lee, Geoffrey M. Voelker, Alec Wolman, Wayne A. Wong, Jean-Loup Baer, Brian N. Bershad, Henry M. Levy

September 1996 **Proceedings of the seventh international conference on Architectural support for programming languages and operating systems**, Volume 31 , 30 Issue 9 , 5

Full text available: pdf(1.17 MB) Additional Information: full citation, abstract, references, citings, index terms

Interpreted languages have become increasingly popular due to demands for rapid program development, ease of use, portability, and safety. Beyond the general impression that they are "slow," however, little has been documented about the performance of interpreters as a class of applications.This paper examines interpreter performance by measuring and analyzing interpreters from both software and hardware perspectives. As examples, we measure the MIPSI, Java, Perl, and Tcl interpreters running an ...

20 A formal protection model of security in centralized, parallel, and distributed systems

Glenn S. Benson, Ian F. Akyildiz, William F. Appelbe

August 1990 **ACM Transactions on Computer Systems (TOCS)**, Volume 8 Issue 3

Full text available: pdf(2.17 MB) Additional Information: full citation, abstract, references, citings, index terms, review

One way to show that a system is not secure is to demonstrate that a malicious or mistake-prone user or program can break security by causing the system to reach a nonsecure state. A fundamental aspect of a security model is a proof that validates that every state reachable from a secure initial state is secure. A sequential security model assumes that every command that acts as a state transition executes sequentially, while a concurrent security model assumes that multiple commands execut ...

Keywords: access control, concurrency control, distributed system security, operating system security, protection model

Terms used **reconfigurable cell** and **instruction** or **command** Found **9,346** of **154,226**

Sort results by relevance
Display results expanded form
Save results to a Binder
Search Tips
Open results in a new window
Try an Advanced Search
Try this search in The ACM Guide

Results 21 - 40 of 200 Result page: previous 1 **2** 3 4 5 6 7 8 9 10 next
Best 200 shown Relevance scale

21 The Vector-Thread Architecture
Ronny Krashinsky, Christopher Batten, Mark Hampton, Steve Gerding, Brian Pharris, Jared Casper, Krste Asanovic
March 2004 **ACM SIGARCH Computer Architecture News , Proceedings of the 31st annual international symposium on Computer architecture - Volume 00**, Volume 32 Issue 2
Full text available: pdf(317.13 KB) Additional Information: full citation, abstract

The vector-thread (VT) architectural paradigm unifies the vectorand multithreaded compute models. The VT abstraction providesthe programmer with a control processor and a vector of virtualprocessors (VPs). The control processor can use vector-fetch commandsto broadcast instructions to all the VPs or each VP can usethread-fetches to direct its own control flow. A seamless intermixingof the vector and threaded control mechanisms allows a VT architectureto flexibly and compactly encode application ...

22 Using emulators as vehicles for instruction in systems programming: prospective consideration
Amos O. Olagunju, Elvis Borders
February 1987 **ACM SIGCSE Bulletin , Proceedings of the eighteenth SIGCSE technical symposium on Computer science education**, Volume 19 Issue 1
Full text available: pdf(332.12 KB) Additional Information: full citation, abstract, references, index terms

Microprogramming is a technique for implementing machine language instruction sets —it is critical in today's computer architectures and operating systems. An emulator is a set of microprograms that implements the architecture of one machine on another; microprogramming is often used in emulation to make one computer system appear as if it were another. This paper presents the architecture for a microprogrammed computer system, the VSEM. The simulated virtual computer system, its moni ...

23 Hypervisor-based fault tolerance
Thomas C. Bressoud, Fred B. Schneider
February 1996 **ACM Transactions on Computer Systems (TOCS)**, Volume 14 Issue 1
Full text available: pdf(1.89 MB) Additional Information: full citation, abstract, references, citings, index terms

Protocols to implement a fault-tolerant computing system are described. These protocols augment the hypervisor of a virtual-machine manager and coordinate a primary virtual machine with its backup. No modifications to the hardware, operating system, or application programs are required. A prototype system was constructed for HP's PA-RISC instruction-set architecture. Even though the prototype was not carefully tuned, it ran programs about a factor of 2 slower than a bare machine would.
Keywords: fault-tolerant computing system, primary/backup approach, virtual-machine manager

24 A PDP-8 emulator program
Brian J. Shelburne
March 2002 **Journal on Educational Resources in Computing (JERIC)**, Volume 2 Issue 1
Full text available: pdf(270.03 KB) Additional Information: full citation, abstract, references, citings, index terms

The clean, simple, and elegant architecture of the classic PDP-8 makes it an ideal candidate for studying concepts in computer organization. The PDP-8 emulator program allows a user to write, edit, assemble, debug, trace, and execute PDP-8 machine code and PDP-8 assembler language programs. With it, the user can obtain a feel for the PDP-8. The PDP-8 emulator program includes a simple built-in text editor which is used to write and edit PDP-8 assembler language programs, an assembler to translat ...

Keywords: Computer architecture simulator, education

25 The IX supercomputer for knowledge based systems

Tetsuya Higuchi, Tatsumi Furuya, Hiroyuki Kusumoto, Ken'ichi Handa, Akio Kokubu
November 1999 **Proceedings of 1986 ACM Fall joint computer conference**

Full text available: pdf(956.81 KB) Additional Information: full citation, references, index terms

26 Lunar orbiter command and telemetry data handling system at deep space stations
E. Knutson, L. Holgersen, D. R. Merrill
January 1966 **Proceedings of the 1966 21st national conference**

Full text available: pdf(1.96 MB) Additional Information: full citation, abstract, references, index terms

The Lunar Orbiter will provide extensive photographic exploration of the lunar surface to aid in the selection of possible landing areas for Project Apollo manned landing mission. The Lunar Orbiter project* is one of the lunar and planetary programs directed by the NASA Langley Research Center. The Boeing Company is the prime spacecraft contractor. There will be five flight spacecraft and three ground test spacecraft. The first flight is scheduled in the middle of 1966. The Lunar ...

27 The reduction of branch instruction execution overhead using structured control flow
Robert G. Wedig, Marc A. Rose
January 1984 **ACM SIGARCH Computer Architecture News , Proceedings of the 11th annual international symposium on Computer architecture**, Volume 12 Issue 3

Full text available: pdf(707.36 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper presents a technique for specifying change of control (e.g. branch) commands at a sequential processor's macroinstruction set level. It is shown that by representing high level language (HLL) control statements with special machine language instructions, the usual delays associated with control flow changes can be reduced. Preserving the HLL control flow information increases performance by reducing both the number of executed branches and pipeline breaks.

28 An experimental investigation of the interactive effects of interface style, instructions, and task familiarity on user performance
Kai H. Lim, Izak Benbasat, Peter A. Todd
March 1996 **ACM Transactions on Computer-Human Interaction (TOCHI)**, Volume 3 Issue 1

Full text available: pdf(2.19 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Norman proposed a model describing the sequence of user activities involved in human-computer interaction. Through this model, Norman provides a rationale for why direct-manipulation interfaces may be preferred to other design alternatives. Based on action identification theory we developed several hypotheses about the operations of Norman's model and tested them in a laboratory experiment. The results show that users of a direct-manipulation interface and a menu-based inte ...

29 Instruction scheduling and executable editing
Eric Schnarr, James R. Larus
December 1996 **Proceedings of the 29th annual ACM/IEEE international symposium on Microarchitecture**

Full text available: pdf(1.10 MB) Additional Information: full citation, abstract, references, citings, index terms

Modern microprocessors offer more instruction-level parallelism than most programs and compilers can currently exploit. The resulting disparity between a machine's peak and actual performance, while frustrating for computer architects and chip manufacturers, opens the exciting possibility of low-cost instrumentation for measurement, simulation, or emulation. Instrumentation code that executes in previously unused processor cycles is effectively hidden. On two superscalar SPARC processors, a simp ...

30 Expressing rhetorical relations in instructional text: a case study of the purpose relation
Keith Vander Linden, James H. Martin
March 1995 **Computational Linguistics**, Volume 21 Issue 1

Full text available: pdf(2.12 MB) Publisher Site Additional Information: full citation, abstract, references, citings

Natural language provides an extensive set of lexical and grammatical forms for expressing concepts, a set from which writers choose the particular form that they feel will produce the most effective expression given the communicative context. An important task of the text generation researcher is to specify both the range of these forms and the contexts in which they are used. This paper addresses this issue in the context of the expression of procedural relations between actions in instruction ...

31 Dynamically altering agent behaviors using natural language instructions
Rama Bindiganavale, William Schuler, Jan M. Allbeck, Norman I. Badler, Aravind K. Joshi, Martha Palmer
June 2000 **Proceedings of the fourth international conference on Autonomous agents**

Full text available: pdf(1.05 MB) Additional Information: full citation, references, citings, index terms

Keywords: autonomous agents, natural language processing, smart avatars, virtual environments

[32] Providing a laboratory for instruction set design

Rosalee Nerheim-Wolfe

March 1992 **ACM SIGCSE Bulletin , Proceedings of the twenty-third SIGCSE technical symposium on Computer science education**, Volume 24 Issue 1

Full text available: pdf(399.09 KB) Additional Information: full citation, abstract, references, index terms

Computer architecture classes do not provide students with laboratory experience in the design of instruction set architectures. Projects that compare designs have not been possible due to a lack of support software. The design and evaluation of a new instruction set requires an assembler, a symbolic debugger, and a statistics gatherer. Every new instruction set requires changes to all three programs. It would be unrealistic to expect that either students or instructor would (re)write such ...

[33] Systems, platforms, and applications: Efficient code distribution in wireless sensor networks

Niels Reijers, Koen Langendoen

September 2003 **Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications**

Full text available: pdf(245.91 KB) Additional Information: full citation, abstract, references, citings, index terms

The need to reprogramme a wireless sensor network may arise from changing application requirements, bug fixes, or during the application development cycle. Once deployed, it will be impractical at best to reach each individual node. Thus, a scheme is required to wirelessly reprogramme the nodes. We present an energy-efficient code distribution scheme to wirelessly update the code running in a sensor network. Energy is saved by distributing only the changes to the currently running code. The new ...

Keywords: code distribution, compression, reprogramming, sensor networks, string distance, wireless

[34] A generalized system for university mathematics instruction

Robert L. Smith, Lee H. Blaine

February 1976 **Proceedings of the ACM SIGCSE-SIGCUE technical symposium on Computer science and education**, Volume 8 , 2 Issue 1 , SI

Full text available: pdf(617.69 KB) Additional Information: full citation, abstract, references, index terms

EXCHECK is a system for developing mathematically-based CAI courses. It is currently being used at Stanford University to teach a college-credit course in axiomatic set theory The design of this system had several goals. First, we wanted an instructional system that would provide a semantic base for our work on processing natural language and computer-generated audio. Axiomatic mathematics fits this description in that the underlying semantics is relatively well understood, but m ...

[35] Regular contributions: More enhancements of the simplescalar tool set

Naraig Manjikian

September 2001 **ACM SIGARCH Computer Architecture News**, Volume 29 Issue 4

Full text available: pdf(709.36 KB) Additional Information: full citation, abstract, references

An earlier paper described enhancements to the SimpleScalar tool set for functional multiprocessor simulation and visualization of cache coherence, and the software was made available at http://www.simplescalar.org. This paper describes additional enhancements to the SimpleScalar tool set. The enhancements include memory access visualization for uniprocessor and multiprocessor simulation, mnltiprocessor enhancement of the DLite! debugger that is included with SimpleScalar, modifications to the G ...

[36] Firefighter command training virtual environment

Tazama U. St. Julien, Chris D. Shaw

October 2003 **Proceedings of the 2003 conference on Diversity in computing**

Full text available: pdf(227.15 KB) Additional Information: full citation, abstract, references, index terms

The Firefighter Command Training Virtual Environment is being developed at Georgia Tech in collaboration with the Atlanta Fire Department. The VE allows the user to: navigate around the environment, viewing a house on fire from any angle; command firefighters and watch them execute those commands; and see realistic fire and smoke behavior reacting to changes in the environment. The VE user is a commanding officer trainee who instructs teams of virtual firefighters to perform different actions to ...

Keywords: simulation, training, virtual reality

[37] Performance enhancements to a relational database system

Michael Stonebraker, John Woodfill, Jeff Ranstrom, Marguerite Murphy, Marc Meyer, Eric Allman

June 1983 **ACM Transactions on Database Systems (TODS)**, Volume 8 Issue 2

Full text available: pdf(1.33 MB) Additional Information: full citation, abstract, references, citings, index terms

In this paper we examine four performance enhancements to a database management system: dynamic compilation, microcoded routines, a special-purpose file system, and a special-purpose operating system. All were examined in the context of the INGRES database management system. Benchmark timings that are included suggest the attractiveness of dynamic compilation and a special-purpose file system. Microcode and a special-purpose operating system are analyzed and appear to be of more limited uti ...

Keywords: compiled query languages, database performance, file systems for databases, microcode

38 A tightly-coupled processor-network interface

Dana S. Henry, Christopher F. Joerg

September 1992 **ACM SIGPLAN Notices , Proceedings of the fifth international conference on Architectural support for programming languages and operating systems**, Volume 27 Issue 9

Full text available: pdf(1.41 MB) Additional Information: full citation, references, citings, index terms

39 Writing instructional materials for computing service courses

Marlene Ménard

September 1990 **ACM SIGDOC Asterisk Journal of Computer Documentation , Proceedings of the 8th annual international conference on Systems documentation**, Volume 14 Issue 4

Full text available: pdf(543.22 KB) Additional Information: full citation, references, index terms

40 A generator of direct manipulation office systems

Scott E. Hudson, Roger King

July 1986 **ACM Transactions on Information Systems (TOIS)**, Volume 4 Issue 2

Full text available: pdf(2.45 MB) Additional Information: full citation, abstract, references, citings, index terms, review

A system for generating direct manipulation office systems is described. In these systems, the user directly manipulates graphical representations of office entities instead of dealing with these entities abstractly through a command language or menu system. These systems employ a new semantic data model to describe office entities. New techniques based on attribute grammars and incremental attribute evaluation are used to implement this data model in an efficient manner. In addition, the s ...

41 Shared cache multiprocessing with pack computers

Stanley Lass

June 1988 **ACM SIGARCH Computer Architecture News**, Volume 16 Issue 3

Full text available: pdf(527.57 KB) Additional Information: full citation, abstract, citings, index terms

A typical cache accesses a block, then returns a selected word. In a pack computer design, the cache accesses and returns a block to the computer. The computer selects the word from the block and also retains blocks for reuse. The retained blocks constitute a software cache.Separate block wide caches for instructions, read data, and write data provide enough bandwidth for several pack computers to execute several instructions per cache cycle time.The concept of a pack (bundle of information) is ...

42 Computer Software and Copyright

Calvin N. Mooers

January 1975 **ACM Computing Surveys (CSUR)**, Volume 7 Issue 1

Full text available: pdf(2.63 MB) Additional Information: full citation, references, citings, index terms

43 IP Design and Reuse: Soft-cores generation by instruction set analysis

Alessandro Fin, Franco Fummi, Giovanni Perbellini

September 2001 **Proceedings of the 14th international symposium on Systems synthesis**

Full text available: pdf(102.59 KB) Additional Information: full citation, abstract, references, index terms

The popularity of *Soft Cores* is rapidly increasing. Their integration in a design requires their synthesis and optimization, thus their application is more complex than the use of *Hard Cores*. However, *Soft Cores* can be customized to the design constraints, thus promising to lead to more efficient designs. The decision of using *Soft* or *Hard Cores}* is difficult since it is a trade-off between performance, cost and design time. The parametrization methodology pres ...

44 On pointers versus addresses

Amir M. Ben-Amram, Zvi Galil

July 1992 **Journal of the ACM (JACM)**, Volume 39 Issue 3

Full text available: pdf(2.30 MB) Additional Information: full citation, references, citings, index terms

Keywords: incompressibility, pointer structures, random access memory

45 Controlled graphs and instructions

Guy Boulaye, Louise Jones

September 1973 **Conference record of the 6th annual workshop on Microprogramming**

Full text available: pdf(424.58 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper is concerned with controlled graphs, a graph-theoretic formalism which was developed from the study of microprogramming. In this formalism, a microinstruction is equivalent to a subgraph of the graph representing the (microprogrammed) computer, and the problem of designing a micro-instruction is simply that of finding an appropriate subgraph. However, this idea can be generalized and applied to other networks which are characterized by flows of discrete elements. The results of o ...

46 Global register allocation at link time

David W. Wall
July 1986 **ACM SIGPLAN Notices , Proceedings of the 1986 SIGPLAN symposium on Compiler contruction**, Volume 21 Issue 7

Full text available: pdf(1.25 MB) Additional Information: full citation, abstract, references, citings, index terms

In previous work in global register allocation, the compiler colors a conflict graph constructed from liveness dataflow information, in order to allocate the same register to many variables that are not simultaneously live. If two procedures are in separately compiled modules, however, the compiler must do this allocation separately for each procedure. As a result, the two procedures might use different registers for the same global, or the same register for different locals.We c ...

47 An interactive simulator for microprogram development

Ning-San Chang, Timothy Mulrooney, Nelson Weiderman
March 1978 **Proceedings of the 11th annual symposium on Simulation**

Full text available: pdf(650.04 KB) Additional Information: full citation, abstract, references, index terms

MICROSIM is an interactive program which simulates the Data General ECLIPSE microprogramming feature. It interprets microprograms written in the ECLIPSE microinstruction's format and produces results, error messages and complete (or specific) trace information according to the user's commands. The interactive interpreter can be used as a self-contained microprogramming interpreter or as a debugging tool to allow ECLIPSE users to develop their own microroutines.

48 BASICI—a simple computer to introduce computer organization and assembler language programming

Donald S. Miller, Bruce R. Millard
February 1982 **ACM SIGCSE Bulletin , Proceedings of the thirteenth SIGCSE technical symposium on Computer science education**, Volume 14 Issue 1

Full text available: pdf(791.01 KB) Additional Information: full citation, abstract, references, citings, index terms

BASICI is a simple interactive assembler-loader/interpreter which has been used as an instructional tool for the introductory course in computer organization and assembler language programming offered by the Computer Science Department at Washington State University. Both "hardware" and software are organized so as to emphasize basic concepts and to eliminate the confusion which occurs when these concepts are first introduced surrounded by the myriad of machine and assembler lan ...

49 Cache Refill/Access Decoupling for Vector Machines

Christopher Batten, Ronny Krashinsky, Steve Gerding, Krste Asanovic
December 2004 **Proceedings of the 37th International Symposium on Microarchitecture**

Full text available: pdf(319.32 KB) Additional Information: full citation, abstract

Vector processors often use a cache to exploit temporal locality and reduce memory bandwidth demands, but then require expensive logic to track large numbers of outstanding cache misses to sustain peak bandwidth from memory. We present refill/access decoupling, which augments the vector processor with a Vector Refill Unit (VRU) to quickly pre-execute vector memory commands and issue any needed cache line refills ahead of regular execution. The VRU reduces costs by eliminating much of the outstan ...

50 Analysis of hardware and software approaches to embedded in-circuit emulation of microprocessors

Hsin-Ming Chen, Chung-Fu Kao, Ing-Jer Huang
January 2002 **Australian Computer Science Communications , Proceedings of the seventh Asia-Pacific conference on Computer systems architecture - Volume 6**, Volume 24 Issue 3

Full text available: pdf(665.32 KB) Additional Information: full citation, abstract, references, index terms

This paper investigates various approaches to embed the functionality of in-circuit emulation (ICE) into microprocessor cores in SoC (System-On-Chip) chips. Three styles of ICE's (hardware-oriented, software-oriented and hybrid) are defined and implemented. They are integrated with a synthesizable ARM7 microprocessor core and synthesized to gate level to quantitatively analyze and compare their performance, cost and debugging features.

51 Minimizing Drum Latency Time

Donald E. Knuth
April 1961 **Journal of the ACM (JACM)**, Volume 8 Issue 2

Full text available: pdf(1.75 MB) Additional Information: full citation, references, citings, index terms

52 A digital video display system implemented on a KIM-1 microcomputer

N. Solntseff, M. D. Drummond
September 1980 **Proceedings of the 3rd ACM SIGSMALL symposium and the first SIGPC symposium on Small systems**

Full text available: pdf(623.80 KB) Additional Information: full citation, abstract, references, index terms

The "microelectronic revolution" and the accompanying decrease in the cost of semiconductor memory has

increased the availability of raster-scan graphical displays, yet, as pointed out in a recent survey [BAE79], the implementation of graphics software for raster-scan systems has lagged behind that for random-scan ones. The aim of the work described in the present paper has been to apply random-scan techniques to a system employing a relatively inexpensive raster-scan device. Th ...

Keywords: Computer animation, Digital video graphics, Display file, Display file processor, Low-cost graphics, Random-scan graphics

53 Simulation coverage and generation for verification: Coverage directed test generation for functional verification using bayesian networks

Shai Fine, Avi Ziv

June 2003 **Proceedings of the 40th conference on Design automation**

Full text available: pdf(162.58 KB) Additional Information: full citation, abstract, references, citings, index terms

Functional verification is widely acknowledged as the bottleneck in the hardware design cycle. This paper addresses one of the main challenges of simulation based verification (or dynamic verification), by providing a new approach for *Coverage Directed Test Generation* (CDG). This approach is based on Bayesian networks and computer learning techniques. It provides an efficient way for closing a feedback loop from the coverage domain back to a generator that produces new stimuli to the test ...

Keywords: bayesian networks, coverage analysis, functional verification

54 Experience with a software-defined machine architecture

David W. Wall

May 1992 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 14 Issue 3

Full text available: pdf(2.86 MB) Additional Information: full citation, abstract, references, citings, index terms, review

We have built a system in which the compiler back end and the linker work together to present an abstract machine at a considerably higher level than the actual machine. The intermediate language translated by the back end is the target language of all high-level compilers and is also the only assembly language generally available. This lets us do intermodule register allocation, which would be harder if some of the code in the program had come from a traditional assembler, out of sight of ...

Keywords: RISC, graph coloring, intermediate language, interprocedural, optimization, pipeline scheduling, profiling, register allocation, register windows

55 The SLANG system

R. A. Sibley

January 1961 **Communications of the ACM**, Volume 4 Issue 1

Full text available: pdf(1.57 MB) Additional Information: full citation, references, citings, index terms

56 Microcode development tools for a capability-based processor

J. Rosenberg, D. A. Abramson

December 1986 **ACM SIGMICRO Newsletter , Proceedings of the 19th annual workshop on Microprogramming**, Volume 17 Issue 4

Full text available: pdf(713.26 KB) Additional Information: full citation, abstract, references, index terms

The MONADS-PC System is a capability-based computer constructed around a microprogrammed processor designed and implemented at Monash University in Melbourne, Australia. This paper describes a set of tools developed as part of this project in order to simplify the implementation and testing of the microcode and to ensure consistency between the microcode and macro-level machine instruction set. Although some of these tools are at present machine specific, the paper demonstrates that they co ...

57 Microinstruction sequencing and structured microprogramming

Louise H. Jones

September 1974 **Conference record of the 7th annual workshop on Microprogramming**

Full text available: pdf(651.54 KB) Additional Information: full citation, abstract, references, citings, index terms

The purpose of this paper is to explore the relation between the sequencing functions of microprogrammable computers and the implementation of the control constructs of flowchartable logic with modular microcode. So far, there has been little or no discussion of this topic in the literature [8]. Sequencing functions for various microprogrammable processors are described in Section 2; the implementation of the Mills' control constructs using various sets ofsequencing functions is discussed i ...

58 Poor man's watchpoints

Max Copperman, Jeff Thomas

January 1995 **ACM SIGPLAN Notices**, Volume 30 Issue 1

Full text available: pdf(772.87 KB) Additional Information: full citation, abstract, citings, index terms

Bugs that result from corruption of program data can be very difficult to track down without specialized help from a debugger. If the debugger cannot help the user find the point at which data gets corrupted, the user may have a long iterative debugging task. If the debugger is able to stop execution of the program at the point where data gets corrupted, as with watchpoints (also known as data breakpoints), it may be a very simple task to find a data corruption bug. In this paper, we discuss a m ...

59 Using functor categories to generate intermediate code

John C. Reynolds

January 1995 **Proceedings of the 22nd ACM SIGPLAN-SIGACT symposium on Principles of programming languages**

Full text available: pdf(1.18 MB) Additional Information: full citation, abstract, references, citings, index terms

In the early 80's Oles and Reynolds devised a semantic model of Algol-like languages using a category of functors from a category of store shapes to the category of predomains. Here we will show how a variant of this idea can be used to define the translation of an Algol-like language to intermediate code in a uniform way that avoids unnecessary temporary variables, provides control-flow translation of boolean expressions, permits online expansion of procedures, and minimizes the storage ov ...

60 A taxonomy of computer program security flaws

Carl E. Landwehr, Alan R. Bull, John P. McDermott, William S. Choi

September 1994 **ACM Computing Surveys (CSUR)**, Volume 26 Issue 3

Full text available: pdf(3.81 MB) Additional Information: full citation, abstract, references, citings, index terms, review

An organized record of actual flaws can be useful to computer system designers, programmers, analysts, administrators, and users. This survey provides a taxonomy for computer program security flaws, with an Appendix that documents 50 actual security flaws. These flaws have all been described previously in the open literature, but in widely separated places. For those new to the field of computer security, they provide a good introduction to the characteristics of security flaws and how they ...

Keywords: error/defect classification, security flaw, taxonomy

Terms used **reconfigurable cell** or **FGPA** and **instruction** or **command**

Found **15,767** of **154,226**

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown

Relevance scale

1 The silicon palimpsest: a programming model for electrically reconfigurable processors

Charles Johnsen, David L. Fox

March 1991 **Proceedings of the second and third annual workshops on Forth**

Full text available: pdf(1.26 MB) Additional Information: full citation, references, index terms

2 Reconfigurable computing: a survey of systems and software

Katherine Compton, Scott Hauck

June 2002 **ACM Computing Surveys (CSUR)**, Volume 34 Issue 2

Full text available: pdf(710.56 KB) Additional Information: full citation, abstract, references, citings, index terms, review

Due to its potential to greatly accelerate a wide variety of applications, reconfigurable computing has become a subject of a great deal of research. Its key feature is the ability to perform computations in hardware to increase performance, while retaining much of the flexibility of a software solution. In this survey, we explore the hardware aspects of reconfigurable computing machines, from single chip architectures to multi-chip systems, including internal structures and external coupling. W ...

Keywords: Automatic design, FPGA, field-programmable, manual design, reconfigurable architectures, reconfigurable computing, reconfigurable systems

3 Configuration cloning: exploiting regularity in dynamic DSP architectures

S. R. Park, W. Burleson

February 1999 **Proceedings of the 1999 ACM/SIGDA seventh international symposium on Field programmable gate arrays**

Full text available: pdf(1.72 MB) Additional Information: full citation, references, citings, index terms

4 Special issue: AI in engineering

D. Sriram, R. Joobbani

January 1985 **ACM SIGART Bulletin**, Issue 91

Full text available: pdf(8.79 MB) Additional Information: full citation, abstract

The papers in this special issue were compiled from responses to the announcement in the July 1984 issue of the SIGART newsletter and notices posted over the ARPAnet. The interest being shown in this area is reflected in the sixty papers received from over six countries. About half the papers were received over the computer network.

5 Associative and Parallel Processors

Kenneth J. Thurber, Leon D. Wald

December 1975 **ACM Computing Surveys (CSUR)**, Volume 7 Issue 4

Full text available: pdf(2.62 MB) Additional Information: full citation, references, citings, index terms

6 The effect of reconfigurable units in superscalar processors

Jorge E. Carrillo, Paul Chow

February 2001 **Proceedings of the 2001 ACM/SIGDA ninth international symposium on Field programmable gate arrays**

Full text available: pdf(208.11 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper describes OneChip, a third generation reconfigurable processor architecture that integrates a Reconfigurable Functional Unit (RFU) into a superscalar Reduced Instruction Set Computer (RISC) processor's pipeline. The architecture allows dynamic scheduling and dynamic reconfiguration. It also provides support for pre-loading configurations and for Least Recently Used (LRU) configuration management.To evaluate the performance of the OneChip architecture, several off-the-s ...

Keywords: OneChip, reconfigurable processors, superscalar processors

7 Delivering acceleration: the potential for increased HPC application performance using reconfigurable logic

David Caliga, David Peter Barker

November 2001 **Proceedings of the 2001 ACM/IEEE conference on Supercomputing (CDROM)**

Full text available: pdf(396.61 KB) Additional Information: full citation, abstract, references, index terms

SRC Computers, Inc. has integrated adaptive computing into its SRC-6 high-end server, incorporating reconfigurable processors as peers to the microprocessors. Performance improvements resulting from reconfigurable computing can provide orders of magnitude speedups for a wide variety of algorithms. Reconfigurable logic in Field Programmable Gate Arrays (FPGAs) has shown great advantage to date in special purpose applications and specialty hardware. SRC Computers is working to bring this technolog ...

Keywords: FPGA, reconfigurable computing

8 Associative Processor Architecture—a Survey

S. S. Yau, H. S. Fung

January 1977 **ACM Computing Surveys (CSUR)**, Volume 9 Issue 1

Full text available: pdf(1.87 MB) Additional Information: full citation, references, citings, index terms

9 The space shuttle primary computer system

Alfred Spector, David Gifford

September 1984 **Communications of the ACM**, Volume 27 Issue 9

Full text available: pdf(5.34 MB) Additional Information: full citation, references, citings, index terms

Keywords: PASS, avionics system, space shuttle

10 Poster session: On hiding latency in reconfigurable systems: the case of merge-sort for an FPGA-based system

Hossam ElGindy, George Ferizis

February 2003 **Proceedings of the 2003 ACM/SIGDA eleventh international symposium on Field programmable gate arrays**

Full text available: pdf(187.05 KB) Additional Information: full citation, abstract

Recursive solutions are effective software techniques that are difficult to map into hardware due to their dependency on input size and data values. As a result, most high-level design tools do not allow for recursive calls. In this paper we present a technique for mapping the merge-sort algorithm, as a case study, into a reconfigurable system. Our mapping employs an on-line prediction method to reconfigure the necessary hardware only when the need arises, and to hide the reconfiguration delay. ...

11 Microservers: a new memory semantics for massively parallel computing

Jay B. Brockman, Peter M. Kogge, Thomas L. Sterling, Vincent W. Freeh, Shannon K. Kuntz

May 1999 **Proceedings of the 13th international conference on Supercomputing**

Full text available: pdf(1.40 MB) Additional Information: full citation, references, citings, index terms

Keywords: massively parallel, microserver, processing-in-memory

12 Poster session: A SC-based novel configurable analog cell

Binlin Guo, Jiarong Tong

February 2003 **Proceedings of the 2003 ACM/SIGDA eleventh international symposium on Field programmable gate arrays**

Full text available: Additional Information:

pdf(187.05 KB) full citation, abstract

This paper presents a high performance Configurable Analog Cell (CAC) which is made up of a Basic Configurable Analog Cell (BCAC) and a digital converter block. The CAC can be used either for Field Programmable Analog Array (FPAA) or for Field Programmable Digital-Analog Mixed Array (FPMA). The BCAC include three innovative Programmable Switch Blocks (PSBs), three Programmable Capacitor Arrays (PCAs), and an amplifier. PSB and PCA can be programmed to generate many equivalent components. In addi ...

13 Communicating logic: an alternative embedded stream processing paradigm
Norbert Imlig, Ryusuke Konishi, Tsunemichi Shiozawa, Kiyoshi Oguri, Kouichi Nagami, Hideyuki Ito, Minoru Inamori, Hiroshi Nakada
January 2000 **Proceedings of the 2000 conference on Asia South Pacific design automation**
Full text available: pdf(286.93 KB) Additional Information: full citation, references

14 Poster session: Reconfigurable randomized K-way graph partitioning
Fatih Kocan
February 2003 **Proceedings of the 2003 ACM/SIGDA eleventh international symposium on Field programmable gate arrays**
Full text available: pdf(187.05 KB) Additional Information: full citation, abstract

In this paper, a randomized k-way graph partitioning algorithm is mapped onto reconfigurable hardware. The randomized algorithm relies on repetitive running of the same algorithm with different random number sequences to achieve the (near-)optimal solution. The run-time and hardware requirements of this reconfigurable solution per a random number sequence are O(|V|-K) cycles and O(|V|log|V|+|E|) gates and flip-flops, respectively. Performance is improved further at the expense of more hardware b ...

15 Composable ad hoc location-based services for heterogeneous mobile clients
Todd D. Hodes, Randy H. Katz
October 1999 **Wireless Networks**, Volume 5 Issue 5
Full text available: pdf(403.18 KB) Additional Information: full citation, references, citings, index terms

16 Maté: a tiny virtual machine for sensor networks
Philip Levis, David Culler
October 2002 **Proceedings of the 10th international conference on Architectural support for programming languages and operating systems**, Volume 37 , 30 , 36 Issue 10 , 5 , 5
Full text available: pdf(1.22 MB) Additional Information: full citation, abstract, references, citings

Composed of tens of thousands of tiny devices with very limited resources ("motes"), sensor networks are subject to novel systems problems and constraints. The large number of motes in a sensor network means that there will often be some failing nodes; networks must be easy to repopulate. Often there is no feasible method to recharge motes, so energy is a precious resource. Once deployed, a network must be reprogrammable although physically unreachable, and this reprogramming can be a significan ...

17 Supporting systolic and memory communication in iWarp
Shekhar Borkar, Robert Cohn, George Cox, Thomas Gross, H. T. Kung, Monica Lam, Margie Levine, Brian Moore, Wire Moore, Craig Peterson, Jim Susman, Jim Sutton, John Urbanski, Jon Webb
May 1990 **ACM SIGARCH Computer Architecture News , Proceedings of the 17th annual international symposium on Computer Architecture**, Volume 18 Issue 3
Full text available: pdf(2.09 MB) Additional Information: full citation, abstract, references, citings, index terms

iWarp is a parallel architecture developed jointly by Carnegie Mellon University and Intel Corporation. The iWarp communication system supports two widely used interprocessor communication styles: memory communication and systolic communication. This paper describes the rationale, architecture, and implementation for the iWarp communication system. The sending or receiving processor of a message can perform either memory or systolic communication ...

18 Clock rate versus IPC: the end of the road for conventional microarchitectures
Vikas Agarwal, M. S. Hrishikesh, Stephen W. Keckler, Doug Burger
May 2000 **ACM SIGARCH Computer Architecture News , Proceedings of the 27th annual international symposium on Computer architecture**, Volume 28 Issue 2
Full text available: pdf(207.54 KB) Additional Information: full citation, abstract, references, citings, index terms

The doubling of microprocessor performance every three years has been the result of two factors: more transistors per chip and superlinear scali ng of the processor clock with technology generation. Our results show that, due to both diminishing improvements in clock rates and poor wire scaling as semiconductor devices shrink, the achievable performance growth of conventional microarchitectures will slow substantially. In this paper, we describe technology-driven models for wire cap ...

19 Tools: Automated tools to implement and test Internet systems in reconfigurable hardware

John W. Lockwood, Chris Neely, Chris Zuver, Dave Lim

July 2003 **ACM SIGCOMM Computer Communication Review**, Volume 33 Issue 3

Full text available: pdf(1.01 MB) Additional Information: full citation, abstract, references, index terms

Tools have been developed to automatically integrate and test networking systems in reconfigurable hardware. These tools dynamically generate circuits for Field Programmable Gate Arrays (FPGAs). A library of hardware-accelerated modules has been developed that processes Internet Protocol (IP) packets, performs header rule matching, scans pocket payloads, and implements per-flow queueing. Other functions can be added to the library as extensible modules.An integration tool was developed to enable ...

Keywords: Field Programmable Gate Array (FPGA), Internet, firewall, network intrusion detection and prevention, networks, reconfigurable hardware, tools

20 Composable ad-hoc mobile services for universal interaction

Todd D. Hodes, Randy H. Katz, Edouard Servan-Schreiber, Lawrence Rowe

September 1997 **Proceedings of the 3rd annual ACM/IEEE international conference on Mobile computing and networking**

Full text available: pdf(1.86 MB) Additional Information: full citation, references, citings, index terms

21 A Modular Computer With Petri Net Array Control

S. S. Reddi

December 1978 **Proceedings of the 1978 annual conference**

Full text available: pdf(550.67 KB) Additional Information: full citation, abstract, references, index terms

This paper presents a computer system with modular functional units which establishes data and control paths dynamically by means of a programmable Petri net array. The input program control flow and the desired allocation of hardware resorces are specified by means of a Petri net and the array by realizing this net controls the data flow between the functional unit. The system exploits parallelism present in the hardware and the program to the extent specified by the programmer. Some of th ...

Keywords: Modular computers, Parallel computer architecture, Petri nets, Reconfigurable computers

22 Active base stations and nodes for wireless networks

Athanassios Boulis, Paul Lettieri, Mani Srivastava

January 2003 **Wireless Networks**, Volume 9 Issue 1

Full text available: pdf(441.19 KB) Additional Information: full citation, abstract, references, citings, index terms

Mobile and wireless network systems are characterized by a highly time varying and heterogeneous operational environment. For example, the wireless link bandwidth and bit error rate can change due to fading, mobile nodes may have different capabilities, and in the course of its movements a mobile node may visit base stations that provide different sets of services, protocols, and interfaces. Adaptability, in various forms and at various levels of the system, is a key to combating the inherent va ...

Keywords: active networking, base station, reconfigurable hardware, wireless and mobile nodes

23 The case for reconfigurable hardware in wearable computing

Christian Plessl, Rolf Enzler, Herbert Walder, Jan Beutel, Marco Platzner, Lothar Thiele, Gerhard Tröster

October 2003 **Personal and Ubiquitous Computing**, Volume 7 Issue 5

Full text available: pdf(469.92 KB) Additional Information: full citation, abstract, index terms

Wearable computers are embedded into the mobile environment of their users. A design challenge for wearable systems is to combine the high performance required for tasks such as video decoding with the low energy consumption required to maximise battery runtimes and the flexibility demanded by the dynamics of the environment and the applications. In this paper, we demonstrate that reconfigurable hardware technology is able to answer this challenge. We present the concept and the prototype implem ...

Keywords: Body area computing system, Embedded systems, Field-programmable gate arrays, Reconfigurable hardware, Wearable computing

24 A memory coherence technique for online transient error recovery of FPGA configurations

Wei-Je Huang, Edward J. McCluskey

February 2001 **Proceedings of the 2001 ACM/SIGDA ninth international symposium on Field programmable gate arrays**

Full text available: pdf(271.54 KB) Additional Information: full citation, abstract, references, index terms

The partial reconfiguration feature of some of the current-generation Field Programmable Gate Arrays (FPGAs) can improve dependability by detecting and correcting errors in on-chip configuration data. Such an error recovery process can be executed online with minimal interference of user applications. However, because Look-up Tables (LUTs) in Configurable Logic Blocks (CLBs) of FPGAs can also implement memory

modules for user applications, a memory coherence issue arises such that memory ...

Keywords: FPGA, error recovery, fault tolerance, memory coherence

25 The family of concurrent logic programming languages
Ehud Shapiro
September 1989 **ACM Computing Surveys (CSUR)**, Volume 21 Issue 3
Full text available: pdf(9.62 MB) Additional Information: full citation, abstract, references, citings, index terms

Concurrent logic languages are high-level programming languages for parallel and distributed systems that offer a wide range of both known and novel concurrent programming techniques. Being logic programming languages, they preserve many advantages of the abstract logic programming model, including the logical reading of programs and computations, the convenience of representing data structures with logical terms and manipulating them using unification, and the amenability to metaprogrammin ...

26 A Hardware/Software Codesign Method for a General Purpose Reconfigurable Co-Processor
Shinji Kimura, Yasufumi Itou, Makoto Hirao, Katumasa Watanabe, Mitsuteru Yukishita, Akira Nagoya
March 1997 **Proceedings of the 5th International Workshop on Hardware/Software Co-Design**
Full text available: pdf(655.95 KB) Publisher Site Additional Information: full citation, abstract, citings

This paper shows a hardware/software codesign method for a computer system with a reconfigurable co-processor. The reconfigurable co-processor is constructed from FPGA's, internal cache and a control part, and is connected to the system bus of the computer system. This paper shows the architecture of the reconfigurable co-processor, a hardware/software separation method and a co-operation method via the DMA based memory sharing. We also show co-operation examples and the effectiveness of our app ...

Keywords: hardware/software co-operation, , a computer architecture using FPGA, , bus-based reconfigurable co-processor architecture, high-level synthesis and optimization, , C compiler to hardware modules

27 Features: The Inevitability of Reconfigurable Systems
Nick Tredennick, Brion Shimamoto
October 2003 **Queue**, Volume 1 Issue 7
Full text available: pdf(1.64 MB) html (40.12 KB) Additional Information: full citation, index terms

28 Poster session: An FPGA architecture with built-in error correction capability
P. K. Lala, B. Kiran Kumar
February 2003 **Proceedings of the 2003 ACM/SIGDA eleventh international symposium on Field programmable gate arrays**
Full text available: pdf(187.05 KB) Additional Information: full citation, abstract

The use of very deep submicron technology makes VLSI-based digital systems more susceptible to transient or soft errors, and thus compromises their reliability. This paper proposes an FPGA architecture inspired by the human immune system that allows tolerance of transient errors. The architecture is composed of a two-dimensional array of identical functional cells with different genetic codes. These codes are chosen based on the required functions to be performed by the functional cells. An erro ...

29 System level modeling and verification: Embedded systems verification with FGPA-enhanced in-circuit emulator
M. Meerwein, C. Baumgartner, T. Wieja, W. Glauert
September 2000 **Proceedings of the 13th international symposium on System synthesis**
Full text available: pdf(109.43 KB) Additional Information: full citation, abstract, references, citings

In this paper we present a novel coverification concept for embedded microcontrollers that satisfies industrial requirements. Based on a commercially available CPU in-circuit emulator coupled with FPGA boards, it verifies the correctness of an implementation in terms of function and timing within a real-world environment.Using our system, the software engineer can write, test and optimize programs for a chip that is not yet physically existent. In addition the system is used to obtain software m ...

30 Poster session: An automated and power-aware framework for utilization of IP cores in hardware generated from C descriptions targeting FPGAs
Alex Jones, Prith Banerjee
February 2003 **Proceedings of the 2003 ACM/SIGDA eleventh international symposium on Field programmable gate arrays**

Full text available: pdf(187.05 KB) Additional Information: full citation, abstract

Use of hand optimized Intellectual Property (IP) logic cores is prolific in hardware design. While IP cores remain a standard way to utilize the improvement in FPGA technology and contend with time to market pressure through reuse, popularity of tools generating hardware descriptions from high-level languages is also increasing in popularity. PACT HDL combines these two methods within a power-aware framework. The PACT HDL compiler generates power optimized VHDL/Verilog from a C language descript ...

31 Poster session: Using FPGAs for data and reorganization engines: preliminary results for spatial pointer-based data structures

Pedro C. Diniz, Joonseok Park

February 2003 **Proceedings of the 2003 ACM/SIGDA eleventh international symposium on Field programmable gate arrays**

Full text available: pdf(187.05 KB) Additional Information: full citation, abstract

FPGAs have appealing features such as customizable internal and external bandwidth and the ability to exploit vast amounts of fine-grain instruction-level parallelism. In this paper we explore the applicability of these features in using FPGAs as data search and reorganization engines for performing search and reorganization computations over spatial pointer-based data structures for which traditional computing platforms perform poorly. The preliminary experiments, for a set of simple spatial qu ...

32 TENEX, a paged time sharing system for the PDP - 10

Daniel G. Bobrow, Jerry D. Burchfiel, Daniel L. Murphy, Raymond S. Tomlinson

March 1972 **Communications of the ACM**, Volume 15 Issue 3

Full text available: pdf(932.60 KB) Additional Information: full citation, abstract, references, citings

TENEX is a new time sharing system implemented on a DEC PDP-10 augmented by special paging hardware developed at BBN. This report specifies a set of goals which are important for any time sharing system. It describes how the TENEX design and implementation achieve these goals. These include specifications for a powerful multiprocess large memory virtual machine, intimate terminal interaction, comprehensive uniform file and I/O capabilities, and clean flexible system structure. Although the ...

Keywords: PDP-10, TENEX, paging, process structure, scheduling algorithm, time sharing system, virtual machines

33 Disco: running commodity operating systems on scalable multiprocessors

Edouard Bugnion, Scott Devine, Kinshuk Govil, Mendel Rosenblum

November 1997 **ACM Transactions on Computer Systems (TOCS)**, Volume 15 Issue 4

Full text available: pdf(400.76 KB) Additional Information: full citation, abstract, references, citings, index terms, review

In this article we examine the problem of extending modern operating systems to run efficiently on large-scale shared-memory multiprocessors without a large implementation effort. Our approach brings back an idea popular in the 1970s: virtual machine monitors. We use virtual machines to run multiple commodity operating systems on a scalable multiprocessor. This solution addresses many of the challenges facing the system software for these machines. We demonstrate our approach with a prototy ...

Keywords: scalable multiprocessors, virtual machines

34 Poster session: An estimation and exploration methodology from system-level specifications: application to FPGAs

Sebastien Bilavarn, Guy Gogniat, Jean Luc Philippe

February 2003 **Proceedings of the 2003 ACM/SIGDA eleventh international symposium on Field programmable gate arrays**

Full text available: pdf(187.05 KB) Additional Information: full citation, abstract

Rapid evaluation and design space exploration from early specifications are important issues in the design cycle. We propose an original area vs. delay estimation methodology that targets reconfigurable architectures. Two main steps compose the estimation flow: i) structural estimations where architectural solutions are defined at the RT level, this step is technological independent and performs an automatic design space exploration and ii) physical estimations which perform technology mapping t ...

35 Poster session: Lattice adaptive filter implementation for FPGA

Zdenek Pohl, Rudolf Matoušek, Jirí Kadlec, Milan Tichý, Miroslav Lícko

February 2003 **Proceedings of the 2003 ACM/SIGDA eleventh international symposium on Field programmable gate arrays**

Full text available: pdf(187.05 KB) Additional Information: full citation, abstract

Our poster introduces an innovative RLS Lattice filter implementation for FPGAs. The signal processing applications typically require wide numeric range, and that poses a problem when using an FPGA implementation. Our approach is based on arithmetic using logarithmic numeric representation (LNS). The

test application - an adaptive noise canceller - has been optimized for the Xilinx Virtex devices. It consumes roughly 70% of all logic resources of the XCV800 device and all block memory cells. The ...

36 Poster session: Making area-performance tradeoffs at the high level using the AccelFPGA compiler for FPGAs

P. Banerjee, V. Saxena, J. Uribe, M. Haldar, A. Nayak, V. Kim, D. Bagchi, S. Pal, N. Tripathi, R. Anderson

February 2003 **Proceedings of the 2003 ACM/SIGDA eleventh international symposium on Field programmable gate arrays**

Full text available: pdf(187.05 KB) Additional Information: full citation, abstract

Applications such as digital cell phones, 3G wireless receivers, and voice over IP, require DSP functions that are typically mapped onto general purpose DSP processors. With the introduction of advanced FPGA architectures which provide built-in DSP support such as the Xilinx Virtex-II, and the Altera Stratix, a new hardware alternative is available for DSP designers. DSP design has traditionally been divided into algorithm development and hardware/software implementation. The majority of DSP alg ...

37 Poster session: FPGA implementation of a fast Hadamard transformer for WCDMA

Sanat Kamal Bahl, Jim Plusquellic

February 2003 **Proceedings of the 2003 ACM/SIGDA eleventh international symposium on Field programmable gate arrays**

Full text available: pdf(187.05 KB) Additional Information: full citation, abstract

In code division multiple access (CDMA) systems the base station identifies each user in a cell by unique orthogonal (Walsh) codes. The Walsh codes are generated at the transmitter using a Walsh-Hadamard function. A Fast Hadamard Transformer (FHT) is used at the receiver to decode the transmitted codes. The purpose of this study is to design a FHT which utilizes less hardware resources as compared to the existing designs and also suggest means for reducing the input length of the Walsh sequence. ...

38 Poster session: Wireless sensor networks: a power-scalable motion estimation IP for hybrid video coding

Federico Quaglio, Maurzio Martina, Fabrizio Vacca, Guido Masera, Andrea Molino, Gianluca Piccinini, Maurizio Zamboni

February 2003 **Proceedings of the 2003 ACM/SIGDA eleventh international symposium on Field programmable gate arrays**

Full text available: pdf(187.05 KB) Additional Information: full citation, abstract

Wireless Sensor Networks are an emerging phenomenon in the research community. The design and development of network architectures and nodes implementation are fostering many research activities. Due to their wide application fields and pervasive employment possibilities, the investigation of novel classes of wireless sensor nodes is of great concern. In this paper we presented a novel Power-Scalable Motion Estimation IP suitable for video-surveillance over Wireless Sensor Networks. The proposed ...

39 Poster session: Power-aware architectures and circuits for FPGA-based signal processing

Frank Honoré, Ben Calhoun, Anantha Chandrakasan

February 2003 **Proceedings of the 2003 ACM/SIGDA eleventh international symposium on Field programmable gate arrays**

Full text available: pdf(187.05 KB) Additional Information: full citation, abstract

This work showcases a power-aware system design methodology for DSP applications on reconfigurable hardware platforms. In particular, an enhanced FPGA architecture is proposed and analyzed for a deep submicron process technology. These enhancements reduce Configurable Logic Block (CLB) usage for distributed arithmetic implementations of signal processing applications by 50% or more thereby reducing the load on interconnect resources. Multi-Threshold CMOS (MTCMOS) circuit design techniques are ag ...

40 Poster session: A granularity-based classification model for systems-on-a-chip

Stephan Bingemer, Peter Zipf, Manfred Glesner

February 2003 **Proceedings of the 2003 ACM/SIGDA eleventh international symposium on Field programmable gate arrays**

Full text available: pdf(187.05 KB) Additional Information: full citation, abstract

Field-programmable logic has become an increasingly important technology for the design of digital circuits. One interesting point in the field of reconfigurable logic is its classification within the implementation space of other technologies. Such a classification gains importance if FPGA technology becomes an integral part of Systems-on-a-Chip (SoC). The poster discusses an approach to classify technologies based on their granularity. Therefore, a new distinction into homogeneous and heteroge ...